File Transfer: Overview

Category: File Transfers

Here's a general overview of the various file transfer scenarios within the NAS environment, with pointers to related articles.

File Transfers Between Pleiades, Columbia, and Lou

File transferring between NAS systems in the secure enclave (Pleiades, Columbia, and Lou) uses host-based authentication (transparent to users) and is usually straightforward. The following articles provide basic information to help you get started.

- Local File Transfer Commands - cp, cxfscp
- Remote File Transfer Commands - scp, bbftp/bbscp
- File Transfer Between Pleiades and Columbia or Lou
- Transferring Files from the Pleiades Compute Nodes to Lou
- Checking File Integrity

File Transfers between a NAS HECC Host and Your Localhost

Transferring files between a NAS host (such as Pleiades,Columbia, or Lou) and a remote host, such as your local desktop, is more complex. There are multiple factors that you should be aware of:

Which commands to use

Remote File Transfer Commands such as *scp* and *bbftp* and *bbscp* are supported on most NAS high-end computing systems. Depending on the way the transfers are performed, you may need either one or both of the client and server software of scp and/or bbftp or the bbscp script installed on your localhost.

- **Transfer Rate**

  File transfer rate with scp, especially using scp from versions of Open that SSH are older than 4.7, can be as slow as 2 MB/sec. For transferring large files over a long distance, consider the following:

  - upgrade to the the latest version of OpenSSH
  - apply the HPN-SSH patch to your OpenSSH

- enable compression by adding -C to the scp command-line if the data will compress well
- use bbftp/bbscp

- **Security Issues**
  - With scp, users' authentication information (such as password or passcode) and data are encrypted.
  - With bbftp and bbscp, only the authentication information is encrypted, while data is not.
  - You can use GPG to encrypt your data prior to the transfer.

Where transfer commands are initiated

- **Outbound file transfers**

  When the file transfer command is initiated on a NAS host such as Pleiades, Columbia, or Lou, the transfer does not need to go through SFE[1,2] or Secure Unattended Proxy. This is the easiest way to transfer files from and/or to your site if your localhost is configured to allow the transfer.

  To learn more, see also Outbound File Transfer Examples.

- **Inbound file transfers**

  When the file transfer command is initiated on a remote host such as your local desktop, the transfer must go through either SFE[1,2] or Secure Unattended Proxy.

  - *Going through SFE[1,2]*

    Going through SFE[1,2] requires authentication via your RSA SecurID fob at the time of operation; you will be prompted for your passcode when you issue the file transfer commands, such as *scp*, *bbftp*, or *bbscp*.

    Transfers can be done with one of the following two approaches:

    1. Two steps: Initiate scp from your localhost to SFE[1,2], and then initiate another scp from SFE[1,2] to Columbia, Pleiades or Lou.

       WARNING: Do not store files on the SFEs since space is very limited. Any file transfers though the SFE really should use the SSH pass-through option described next.
    2. One step: Initiate scp, bbftp/bbscp from your localhost to PleiadesmColumbia, or Lou if SSH Passthrough has been set up.

    To learn more, see also Inbound File Transfers through SFEs Examples.

♦ *Going through SUP*

Going through the Secure Unattended Proxy does *not* require SecurID fob authentication at the time of operation. Instead, special "SUP keys" using SecurID authentication must be obtained ahead of time. The "SUP keys" are good for one week and are used automatically to authenticate users for file transfers using scp, bbftp or bbscp issued on a command-line or in a job script.

WARNING: Although users have accounts on the SUP servers, no login session is allowed.

File transfers going through SUP offers multiple benefits over going through the SFEs:

◊ SUP allows the transfer to be unattended; that is, you do not have to type in your password, passphrase, or passcode when the file transfer command is issued. So, file transfers can be done within a script that can be scheduled to run ahead of time. On the other hand, file transfers through the SFEs can not be done in a script.
◊ File transfers through SUP are done in one step, and setting up SSH passthrough is not needed since the SFEs are not involved.
◊ SUP automatically sets some options, such as the port range allowed for bbftp transfers, so that you don't have to set them explicitly. Thus, the syntax for bbftp over SUP is greatly simplified compared to bbftp without SUP.

NOTE: Some sites only allow specific outbound ports; this may cause bbftp to break.
◊

See the article Using the Secure Unattended Proxy (SUP) and the examples there for more information.

- **File staging**

When there are issues (such as a firewall) that hinder the inbound and/or outbound transfers, file staging through DMZFS[1,2] is another option. You can deposit and retrieve files on DMZFS[1,2] by issuing the *scp* or *bbftp* command on either a NAS host or your localhost.

WARNING: The total storage space on DMZs is 2.5TB, shared among all users; files older than 24 hours are removed.

DMZFS[1,2] do not use SecurID fob for authentication. Instead, password or public key authentication is used for file transfers via DMZFS[1,2].

Unattended file transfers can also be done through DMZFS[1,2] if public key authentication has been set up on DMZFS[1,2].

Note, however, that for this purpose, the SUP is preferred as SUP transfers are direct to the end target so do not have the storage restrictions and two step performance limitations of DMZFS when using bbftp/bbscp.

Read File Staging through DMZ File Servers for more information.

NAS Username and Your Local Username

If your NAS username and local username are different, you may have to add the appropriate username in the scp, bbftp or bbscp command-line.

- If you issue the command on your local host, then the username is your NAS username.
- If you issue the command on a NAS host, then the username is your local username.

In the examples shown in the articles Outbound File Transfer Examples and Inbound File Transfers through SFEs Examples, you will find the correct syntax for adding the appropriate username in the file transfer commands.

For inbound file transfers, if you have correctly included your NAS username in the *~/.ssh/config* file of your localhost, you do not have to include the NAS username in the *scp*, *bbftp* or *bbscp* command. A template for the *~/.ssh/config* is available for download.

Check File Integrity Before and After the Transfer

It's a good practice to ensure the integrity of the data before and after the transfer. For more information, see Checking File Integrity.

Tuning your Local System to Improve File Transfer Performance

Some file transfer commands provide options that can be used to improve your transfer rates. For example, enabling compression during file transfers may help in some cases: with bbftp, you can use multiple streams instead of a single stream for better performance. Read Tips for File Transfers for more information.

On the other hand, file transfer performance is also dependent on some system-wide settings. If necessary, ask your local system administrator to look into issues discussed in the following articles:

- TCP Performance Tuning for WAN Transfers
- Optional Advanced Tuning for Linux
- Pittsburgh Supercomputing Center's Enabling High Performance Data Transfers - a properly tuned TCP/IP stack

---

Article ID: 140
Last updated: 16 May, 2011
Data Storage & Transfer -> File Transfers -> File Transfer: Overview
http://www.nas.nasa.gov/hecc/support/kb/entry/140/?ajax=1